

IN THE CLAIMS:

Please amend the claims as follows:

1. (Currently Amended) A method of communicating between a TCP stack, wherein the TCP stack delegates one or more connections to the offload unit, and the TCP stack processes connections that are not delegated or require special processing, and an offload unit, comprising:
 - utilizing a driver as a translator for writing a command including an index corresponding to a delegated connection to an entry in a command ring;
 - indicating by a pointer an owner of the entry, the owner being the TCP stack or the offload unit, wherein every entry in the ring includes a pointer indicating the owner of the entry;
 - reading the command from the entry in the command ring to the offload unit or the TCP stack based on the pointer;
 - executing the command; and
 - writing command specific status to the entry in the command ring by the offload unit, said offload unit indicating to the TCP unit by a pointer the owner of the entry.
2. (Original) The method of claim 1, wherein the command includes a location of a buffer for storing payload data produced by the offload unit.
3. (Original) The method of claim 1, wherein the command includes connection information needed to setup a delegated connection.
4. (Original) The method of claim 1, wherein the command specific status includes a value representing a number of buffers accepted by the offload unit.
5. (Original) The method of claim 1, further comprising:
 - writing a notification descriptor including an index corresponding to a delegated connection to an entry in a notification ring; and

reading the notification descriptor from the entry in the notification ring.

6. (Original) The method of claim 5, wherein the notification descriptor includes one or more notification flags indicating specific information for a connection.

7. (Currently Amended) A method of communicating between a TCP stack and an offload unit, wherein the TCP stack delegates one or more connections to the offload unit, the TCP stack processing connections that are not delegated or require special processing comprising:

utilizing a driver as a translator for writing a notification descriptor to an entry in a notification ring;

indicating by a pointer an owner of the entry, the owner-being one of the TCP stack or the offload unit, wherein every entry in the ring includes a pointer indicating the owner of the entry to the offload unit and the TCP unit;

reading the notification descriptor from the entry in the notification ring;

reading the command from the entry in the command ring to the offload unit or the stack based on the pointer; and

determining connection information for a delegated connection based on the notification descriptor.

8. (Original) The method of claim 7, wherein the notification descriptor includes an index corresponding to the delegated connection.

9. (Original) The method of claim 7, wherein the notification descriptor includes one or more notification flags the delegated connection.

10. (Original) The method of claim 7, wherein the notification descriptor includes a count of received acknowledgements.

11. (Original) The method of claim 8, wherein a flag indicates an acknowledgement threshold was reached on the delegated connection.

12. (Original) The method of claim 8, wherein a flag indicates a duplicate acknowledgement was received on the delegated connection.
13. (Original) The method of claim 8, wherein a flag indicates a sequence number threshold was reached on the delegated connection.
14. (Original) The method of claim 8, wherein a flag indicates at least a portion of frame data received on the delegated connection was uploaded by the offload unit to a legacy buffer.
15. (Original) The method of claim 8, wherein a flag indicates a request for a user buffer for uploading of payload data from the offload unit.
16. (Currently Amended) A system for transmitting commands from a TCP stack wherein the TCP stack delegates one or more connections to the offload unit, the TCP stack processing connections that are not delegated or require special processing, to an offload unit, comprising:
a command ring configured to receive commands written by the TCP stack;
[[and]]
a driver configured as a translator for writing a command including an index corresponding to a delegated connection to an entry in a command ring;
the driver indicating by a pointer an owner of the entry, wherein every entry in the ring includes a pointer indicating the owner of the entry to the offload unit and the TCP unit; and
a command unit the offload unit configured to read commands from the command ring and to process the commands only when the pointer indicates the offload unit is not the owner of the entry.
17. (Original) The system of claim 16, further comprising:

a notification ring configured to receive connection information written by the offload unit and output connection information read by the TCP stack.

18. (Original) The system of claim 16, wherein the offload unit is configured to write command specific status to the command ring.

19. (Original) The system of claim 16, further comprising a transmit descriptor ring configured to transfer transmit buffer information from the TCP stack to the offload unit.

20. (Original) The system of claim 19, wherein the transmit buffer information includes a delegated connection index.

21. (Original) The system of claim 16, further comprising a receive descriptor ring configured to transfer receive buffer information from the TCP stack to the offload unit.

22. (Currently Amended) A system for transmitting connection information from an offload unit to a TCP stack wherein the TCP stack delegates one or more connections to the offload unit, the TCP stack processing connections that are not delegated or require special processing, comprising:

a notification unit within the offload unit configured to write notification descriptors to a notification ring; [[and]]

utilizing a driver as a translator for writing a command including an index corresponding to a delegated connection to an entry in a command ring; indicating by a pointer an owner of the entry, wherein every entry in the ring includes a pointer indicating the owner of the entry to the offload unit and the TCP unit; and

the notification ring configured to output notification descriptors read by the TCP stack only when the pointer indicates the offload unit is not the owner of the entry.

23. (Original) The system of claim 22, wherein the notification unit is configured to set one or more flags, each flag indicating connection information.

24. (Currently Amended) The system of claim ~~[[21]]~~ 22, wherein the notification unit is configured to set a flag when at least a portion of a partially parsed frame is uploaded to a legacy buffer.

25. (Currently Amended) The system of claim ~~[[21]]~~ 22, wherein each notification descriptor includes a sequence number.

26. (Currently Amended) The system of claim ~~[[21]]~~ 22, wherein each notification descriptor includes an acknowledgement number.

27. (Currently Amended) The system of claim ~~[[21]]~~ 22, wherein each notification descriptor includes a count of received acknowledgements.

28. (Currently Amended) A system for communicating between a TCP stack wherein the TCP stack delegates one or more connections to the offload unit, the TCP stack processing connections that are not delegated or require special processing, and an offload unit, comprising:

means for transmitting commands from the TCP stack to the offload unit; ~~[[and]]~~
utilizing a driver as a translator for writing a command including an index corresponding to a delegated connection to an entry in a command ring;
indicating by a pointer an owner of the entry, wherein every entry in the ring includes a pointer indicating the owner of the entry to the offload unit and the TCP unit;
and

means for transmitting notification descriptors from the offload unit to the TCP stack only when the pointer indicates the offload unit is not the owner of the entry.

29. (Original) The system of claim 28, further comprising:

means for transmitting command-specific status from the offload unit to the TCP stack.

30. (Original) The system of claim 28, further comprising:
means for transmitting receive buffer descriptors from the TCP stack to the offload unit.

31. (Original) The system of claim 28, further comprising:
means for transmitting transmit buffer descriptors from the TCP stack to the offload unit.

32. (New) A method as claimed in claim 1, wherein when the TCP stack unites the entry to the command ring, the pointer comprising Command Write Pointer is set to indicate that the entry is owned by the offload unit.

33. (New) A method as claimed in claim 1, wherein when the offload unit writes the entry to the command unit, the pointer comprising command read pointer is set to indicate that the entry is owned by the TCP stack.

34. (New) A method as claimed in claim 33, wherein when the offload unit writes the entry to the command unit, the pointer comprising Command Read Pointer is set to indicate that the entry is owned by the TCP stack.

35. (New) A method as claim in claim 31, wherein at startup the pointer indicates the entries owned by the TCP stack.

36. (New) A method as claimed in claim 34, wherein the Command Read Pointer is not permitted to pass the Command Write Pointer.

37. (New) A method as claims in claim 36, including the step of the offload unit using the ring to notify the TCP stack of offloaded connections needing further processing by the TCP stack.